EDITORIAL

Translational Computer Science for Science and Engineering

Manish Parashar, *University of Utah, Salt Lake City, UT, 84112, USA*

David Abramson, *University of Queensland, Brisbane, QLD, 4072, Australia*

Over the past 50 years, computer science research has contributed enormously to the progress of science and engineering (S&E). In fact, it is difficult to imagine any field of S&E that has not been impacted. Equations lacking closed form solution are routinely solved with sufficient precision using numerical methods. Aircraft fly, reactors operate, cars drive, minerals are mined and processed, all in ways that would not be possible without advanced computing. The increasing availability of data is opening new opportunities for advancing all areas of S&E and can benefit significantly from advances in computer science.

Looking ahead, one can expect dramatic changes in applications across all of S&E. These are driven in part by the increasing availability and scales of experimental and observational data, the ability to model phenomena more holistically across multiple scales and physics, the need for near-real-time data processing and actuation, etc. Complementing these changes in applications, the landscape of technologies and resources is also changing, providing new capabilities while at the same time adding new complexities. This includes the emergence of novel storage technologies and deep memory hierarches, the proliferation of accelerators, the deployment of high-bandwidth/low-latency networks, the increasing capabilities available at the network edges, and the proliferation of cloud services and technologies such as containers.

Advances in computer science are essential to address this growing complexity. These advances span new algorithms and system architectures, new software abstractions and programming systems, new execution paradigms and environments, integration of data-driven methods and AI/ML techniques, etc. The role of computer science research in the continuing advance of applicable methods is critical. For example, systems that were previously science fiction, such as real time language translation and speech recognition, are now available only because of deep computer science research.

Traditionally, computer science research has been split into pure and applied streams—foundational research being more "blue sky" without clear and immediate applications, and applied research being performed when explicit problems need to be solved. Even this distinction has blurred over time because there are many examples of research that appeared foundational and have had enormous applications in time. For example, graph theory is critical for most of the genomic analysis performed routinely these days. However, as the application, computer science, and technology landscapes evolve rapidly, it is becoming important that we closely couple cycles of innovation between computer science and other disciplines.

In contrast to computer science research, the field of medicine has embraced a third mode of enquiry called Translational Research (TR). In TR, the pure and applied methods are augmented with a focus on "translation into practice." TR differs from applied research in which the application phase is intrinsically bound into the research timeline rather than being an optional and separate activity. By doing this the applicability of the work can be fed back into the research itself, increasing the chance that the methods scale and apply in the real world. Translational medicine (TM) talks of research done at the *bench* being translated to the *bedside* in consultation with a *community* of practice. These three aspects of TM guarantee good research is done in the first place and the work is applied in partnership with those who will use it. Importantly, the flow is not unidirectional, and information gained from application is fed back to the *bench* for modification and enhancement. Likewise, successful adoption is improved because the *community* feeds back information into the core research. TM is best known for its application to pharmaceutical research, and the current rapid development of COVID-19 vaccines serves as the most poignant contemporary illustration of the overall process.

1521-9615 © 2021 Crown Copyright
Digital Object Identifier 10.1109/MCSE.2021.3109962
Date of current version 23 September 2021.

In an article in *IEEE Computer* in 2019,[1] we proposed an analogous paradigm for computer science research called Translational Computer Science (TCS). Building on the successful TM framework, we relabeled the TM axes *laboratory*, *locale,* and *community*. In that paper, we outlined the benefits of TCS such as more pragmatic applied research and the reduction in time to adoption. We also distinguished translation from commercialization, the latter which implies a financial angle that may not necessarily be present in all applications of computer science.

We identified a number of reasons that TCS has not been widely adopted to date. Most notable among these was the lack of support for translation in most research funding schemes and reward structures. Traditional grants support both pure and applied research, but it is difficult to translate results with existing funding envelopes and timelines. Further, traditional Ph.D. programs make it difficult for a student to both perform research and translate it in the limited time available. Most computer science publication venues do not place any value on translation, and as a result, papers that report on translational aspects are not that highly regarded. Consequently, traditional metrics suffer if a researcher spends the time on translation as well as the research. Finally, we lack good exemplars, meaning researchers cannot see what is possible.

In a recent issue of the *Journal of Computational Science*,[2] we solicited papers from teams who were managing to perform translational research "against the odds." Our goal was to both highlight the benefits but also draw attention to the inhibitors. Interestingly, across 11 papers all the detractors cited in our original paper appeared, plus an additional one, i.e., sustaining research artefacts such as software products.

In this new CiSE Department, we will solicit papers that both promote TCS and highlight the challenges. Here, we focus on the role in science and engineering applications to demonstrate that supporting translation not only leads to better computer science research, but also aids the field it supports. Articles appearing in this department will focus on the processes used by researchers to translate computer science innovations into impact on one or more S&E domains, as well as the lessons learnt along the way. We expect that the Department leads to changes in our organizations and in the funding agencies to better support translational research over the long term.

REFERENCES

1. D. Abramson and M. Parashar, "Translational research in computer science," *Computer*, vol. 52, no. 9, pp. 16–23, Sep. 2019, doi: 10.1109/MC.2019.2925650.
2. D. Abramson, M. Parashar, and P. Arzberger, "Translation computer science—Overview of the special issue," *J. Comput. Sci.*, vol. 52, 2020, Art. no. 101227. [Online]. Available: https://www.sciencedirect.com/journal/journal-of-computational-science/special-issue/

MANISH PARASHAR is the Director of the Scientific Computing and Imaging (SCI) Institute, the chair in computational science and engineering, and a professor, School of Computing, University of Utah. He is currently on an IPA appointment at the National Science Foundation where he is serving as the Office Director of the NSF Office of Advanced Cyberinfrastructure. He is the Founding Chair of the IEEE Technical Consortium on High Performance Computing, the editor-in-chief of the *IEEE Transactions on Parallel and Distributed Systems*, and serves on the editorial boards and organizing committees of several journals and international conferences and workshops. He is a Fellow of AAAS, ACM, and IEEE. For more information, please visit http://manishparashar.org.

DAVID ABRAMSON is the Director of research computing at the University of Queensland, Brisbane, Australia. He is an associate editor of the *Future Generation Computer Systems Journal* (Elsevier) and the president of the Computing Research and Education Association of Australasia. He is a fellow of the IEEE, Association for Computing Machinery, Australian Academy of Technological Sciences and Engineering, and Australian Computer Society. He received the Ph.D. and D.Sc. degrees in computer science from Monash University, Victoria, Australia. Contact him at david.abramson@uq.edu.au.